Pattern Recognition Letters 51 (2015) 79–85

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

A comparison of two Monte Carlo algorithms for 3D vehicle trajectory reconstruction in roundabouts ☆

Andrea Romanoni [a,*], Lorenzo Mussone [b], Davide Rizzi [a], Matteo Matteucci [a]

[a] Politecnico di Milano, DEIB, Via Ponzio, 34/5, Milano, 20133, Italy
[b] Politecnico di Milano, ABC, Via Bonardi, 9, 20133, Milano, Italy

ARTICLE INFO

Article history:
Received 14 March 2014
Available online 16 September 2014

Keywords:
Vehicular trajectory
3D visual tracking
Monte Carlo smoothing
Model-based tracking

ABSTRACT

Visual vehicular trajectory analysis and reconstruction represent two relevant tasks both for safety and capacity concerns in road transportation. Especially in the presence of roundabouts, the perspective effects on vehicles projection on the image plane can be overcome by reconstructing their 3D positions with a 3D tracking algorithm. In this paper we compare two different Monte Carlo approaches to 3D model-based tracking: the Viterbi algorithm and the Particle Smoother. We tested the algorithms on a simulated dataset and on real data collected in one working roundabout with two different setups (single and multiple cameras). The Viterbi algorithm estimates the Maximum A-Posteriori solution from a sample-based state discretization, but, thanks to its continuous state representation, the Particle Smoother overcomes the Viterbi algorithm showing better performance and accuracy.

1. Introduction

Vehicular monitoring is one of the most relevant research topics in the intelligent transportation systems field. A system capable of estimating vehicle position and its dynamics on the road is important, for instance, to detect infractions as well as road accidents [1], and it could also provide useful information about traffic distribution [2]. In the context of vehicular monitoring, roundabouts represent a uniquely challenging scenario for their complexity, both in terms of vehicular trajectories (which are different between vehicles of the same class and very different between vehicles of different classes) and in terms of simultaneous occlusions of more vehicles, especially occurring in multi-lanes circulatory roadways and with heavy vehicles.

One approach to traffic monitoring is the visual vehicle tracking [3], i.e., the process of recognizing moving objects and estimating their trajectory from a video sequence. Most of the existing visual vehicle tracking systems propose a 2D approach (2D tracking hereafter): these systems identify moving vehicles on the image plane, e.g., by identifying their blobs (see Fig. 1) via background subtraction [4], and they track their trajectories on this plane [5,2]. Although, in some applications, this type of estimate might be sufficient to fully understand the vehicle behavior, in many cases, especially in roundabout intersections, we need to estimate vehicle trajectories with high accuracy and with respect to a 3D world reference system; the latter process is called *3D tracking*.

The straightforward approach to reconstruct a 3D trajectory projects the 2D vehicle positions — approximately the centroids of the blobs estimated with 2D tracking — from the image plane on the road plane as in Fig. 2, where C_{wrong} is the intersection of the centroid viewing ray with the road ground plane, see [6,2]. The main drawbacks of this approach are the high sensitivity to perspective deformations and the effect of the unknown height of vehicles, especially when they are heavy trucks. An example of that latter is reported in Fig. 2 where the estimated center C_{wrong} is far from the real vehicle center C.

To overcome these issues, some researchers have proposed to use 3D model-based tracking algorithms and this is the approach we focus on in this paper. This class of algorithms gives a trajectory estimation in 3D world coordinates by representing the tracked object with one or more models, for instance in our implementation we have used a set of parallelepipeds with variable dimensions and we infer (computationally) which model should be used for the current vehicle.

The two most common approaches to 3D model-based vehicle tracking are named *edge-based* and *region-based*, according to the features used to recognize and track a vehicle. The latter, i.e., the region-based, has shown more flexibility and robustness [7] and for this reason we focus our comparison on this class of algorithms. The most suitable way to deal with region-based 3D tracking is by means of a Monte Carlo estimation [8,9], since it natively applies the concept of hypotheses scoring, very useful when comparing the 3D vehicle model back-projected on the image plane against the region occupied

☆ This paper has been recommended for acceptance by R. Davies.

* Corresponding author. Tel.: +39 02 2399 4031.
E-mail address: andrea.romanoni@polimi.it (A. Romanoni).

http://dx.doi.org/10.1016/j.patrec.2014.09.003
0167-8655/

(a) *Vehicle.* (b) *Blob.*

Fig. 1. Blob extracted through background subtraction.

by the vehicle. Moreover Monte Carlo estimation does not rely on the strong Gaussian and unimodal assumptions of the common Kalman estimation.

An abridged version of this paper was presented in the conference paper [10]. Here we propose an analysis of two Monte Carlo approaches to region-based 3D tracking by comparing a Viterbi algorithm and a Particle Smoother. The former deals with a discrete representation of the state to provide the Maximum A-Posteriori (MAP) estimate, while the latter approximates the MAP solution through its sample-based distribution; at the end of our analysis we show that, thanks to its continuous state representation, the Particle Smoother gives better results and with a lower computational cost. While in [10] we have focused on the relevance of 3D tracking with respect to the 2D one in a roundabout setting, in this paper we focus on the comparison of two 3D tracking algorithms, and we discuss on the reasons why a Monte Carlo approach fits well the region-based 3D tracking.

In Section 2, we present a literature overview on 3D model-based tracking algorithms. In Section 3 we present a (Bayesian smoothing) formalization of the tracking and smoothing problem. Then, in Section 4 we describe the two Monte Carlo tracking algorithms and how they implement the Bayesian smoothing. In the same section we explain also the vehicle model management, the algorithm functioning and how the likelihood is computed in the single and multiple camera cases. In Section 5 we illustrate the experimental results for the two tracking algorithms on simulated and real scenarios, while Section 6 concludes the paper.

2. 3D model-based tracking

The visual tracking process aims at estimating the state of an object from a sequence of images. The classical computer vision tracking involves the estimation of the object position on the image plane, hence it is named 2D tracking. A lot of approaches to 2D tracking have been presented, see [11]; the most successful ones learn the appearance of the object and track it leveraging on the learned description. The main differences among the various algorithms lie on what kind of features they learn and on how they model the tracked object: for instance [12] and [13] learn the color histogram of the object; [14] learn an eigenbasis representation; [15] model the object with SIFT features; and a very recent and successful approach uses sparse coding to represent the objects, see [16].

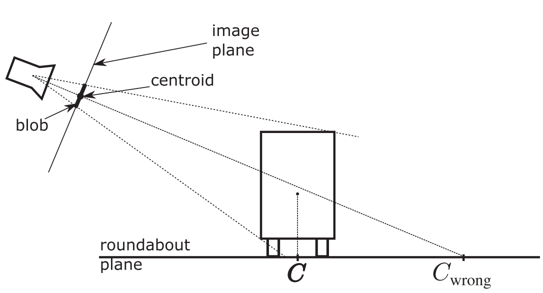

Fig. 2. 2D to 3D projection of vehicle position: the estimate C_{wrong} differs significantly from the real position C.

Another approach to object tracking is named 3D tracking and it estimates the sequence of 3D positions [7]. Especially in the vehicle tracking scenarios this approach represents a widespread method; indeed in the comprehensive review of vehicular trackers [17] most of the analyzed systems estimate the 3D position and usually by means of a 3D model. To simplify the tracking task, all studies in 3D tracking literature assume that the camera calibration is known, see [18]; then, they usually assume the Ground Plane Constraint, i.e., a vehicle always lies on the road plane, and tracking is executed on this plane in order to diminish the vehicle degrees of freedom to be estimated from 6 to 3. Most of these 3D tracking systems make use of an object model, indeed they are called model-based tracking systems. Briefly, vehicle position is estimated, frame-by-frame, by looking for the best model position and orientation which fit the object measure extracted from the images.

The authors in [7] classify the 3D tracking algorithm in: edge-based, region-based, optical flow-based and feature-based. In the vehicle tracking literature the edge-based and region-based represent the most common approaches. They both project the vehicle model from the estimated pose on the image plane, but they differ in the choice of the metrics adopted to evaluate the current estimate.

The *edge-based* algorithms compare the model projection with the image edges; starting from the current estimate, they look for the roto-translation that minimizes the distance between projected segments of the model and the image edges [19,20]. This method has the advantage of being robust to light changes and to image noise, but relevant failures may occur during the minimization step, since the algorithm often stops on local minima.

The *region-based* algorithms compare the model projection with the image region occupied by the tracked object, usually referred to as blob (see Fig. 1), typically extracted by background subtraction [4] or frame-by-frame difference [21]. To estimate the vehicle pose, some region-based algorithms minimize a metric, as for the edge-based case [22], while other algorithms calculate a convenient score for a set of hypothesized model poses [23,8]. The former approach aims at diminishing significantly the number of local minima compared to the edge-based method while the latter almost eliminates them.

Even if the edge-based approach is more robust to light changes and image noise, the region-based one is a more adequate choice for vehicle tracking: it is flexible, since it does not require an exact model of the vehicle; it is robust to the local minima issue; and it relies on background subtraction, a well known module implemented in most video surveillance systems.

Both the edge-based and the region-based approaches usually adopt the Kalman Filter [24,23,25] to perform model-based 3D tracking; but an increasing number of researchers have adopted a Monte Carlo approach [8,9] where the vehicle state is represented by a set of weighted samples.

In the region-based algorithms, the most effective way to deal with the comparison between blob and model projection is the computation of an overlap score; with the Kalman Filter this score cannot be used, and the common solution is to back-project on the road plane the blob centroid, then compare it with the Kalman state prediction. This process has two main limitations: by using just the back-projected blob centroid we neglect a lot of information coming from the blob dimension and shape, moreover we cannot directly compute the fitting of the 3D vehicle model to the measurement. Conversely, a Monte Carlo approach natively weights hypotheses with a likelihood